Proc. 23rd Int. Workshop on Description Logics (DL2010), CEUR-WS 573, Waterloo, Canada, 2010.

Query Answering in the Description Logic $\mathcal{S}^\star$

Meghyn Bienvenu[1], Thomas Eiter[2], Carsten Lutz[1],
Magdalena Ortiz[2], Mantas Šimkus[2]

[1] Fachbereich Mathematik und Informatik
Universität Bremen, Germany
(meghyn|clu)@informatik.uni-bremen.de

[2] Institute of Information Systems
TU Vienna, Austria
(eiter|ortiz|simkus)@kr.tuwien.ac.at

Abstract. We consider the complexity of answering conjunctive queries in the description logic $\mathcal{S}$, i.e., in $\mathcal{ALC}$ extended with transitive roles. While a co-NExpTime lower bound was recently established in [5], the best known upper bound was 2-ExpTime. In this paper, we concentrate on the case where only a single transitive role (and no other role) is present and establish a tight co-NExpTime upper bound.

1 Introduction

Formal ontologies have gained significant importance in the last decade and play an increasing role in a growing number of application areas including the semantic web, ontology-based information integration, and peer-to-peer data management. As a result, ontology formalisms such as description logics (DLs) are nowadays required to offer support for query answering that goes beyond simple taxonomic questions and membership queries. In particular, conjunctive queries (CQs) over instance data play a central role in many applications and have consequently received considerable attention, cf. [11, 6, 9] and references therein and below.

A main aim of recent research has been to identify the potential and limitations of CQ answering in various DLs by mapping out the complexity landscape of this reasoning problem. When concerned with inexpressive DLs such as DL-Lite and $\mathcal{EL}$, one is typically interested in data complexity and efficient implementations based on relational database systems [3, 8]. In expressive DLs, the data complexity is almost always coNP-complete and it is more interesting to study combined complexity. While 2-ExpTime upper bounds for expressive DLs of the $\mathcal{ALC}$ family are known since 1998 [4], lower bounds except ExpTime-hardness (which is trivially inherited from satisfiability) have long been elusive. A first step was made in [7], where *inverse roles* were identified as a source of complexity: CQ answering in plain $\mathcal{ALC}$ remains ExpTime-complete, but goes up to 2-ExpTime-completeness in $\mathcal{ALCI}$. When further extending $\mathcal{ALCI}$ to the popular DL $\mathcal{SHIQ}$, CQ answering remains 2-ExpTime-complete [6].

⋆ This work was partially supported by the Austrian Science Fund (FWF) grant P20840, the EC project OntoRule (IST-2009-231875) and the CONACYT grant 187697.

Interestingly, inverse roles turn out not to be the only source of complexity in $\mathcal{SHIQ}$. In [5], we have shown that transitive roles, which play a central role in many ontologies and are used to represent fundamental relations such as "part of" [10], also increase the complexity of CQ answering. More specifically, CQ answering is co-NExpTime-hard in the DL $\mathcal{S}$, which is $\mathcal{ALC}$ extended with transitive roles and the basic logic of the $\mathcal{SHIQ}$ family, even with only a single transitive role and no other roles (and when the TBox is empty). We have also shown in [5] that if we further add role hierarchies and thus extend $\mathcal{S}$ to $\mathcal{SH}$, CQ answering even becomes 2-ExpTime-complete.

However, the precise complexity of CQ answering in $\mathcal{S}$ has remained open between co-NExpTime and 2-ExpTime. The only existing tight bound (also from [5]) concerns tree-shaped ABoxes, for which CQ answering in $\mathcal{S}$ is only ExpTime-complete (which is remarkable because previously known lower bounds for CQ answering in DLs did not rely on the ABox structure). In this paper, we present ongoing work on CQ answering in $\mathcal{S}$ and show that, in the presence of only a single transitive role and no other role, CQ answering in $\mathcal{S}$ is in co-NExpTime, thus co-NExpTime-complete. This result is interesting for two reasons. First, co-NExpTime is an unusual complexity class for CQ answering in expressive DLs as all previous extensions of $\mathcal{ALC}$ have turned out to be complete for a deterministic time complexity class; the only exception is a co-NExpTime result for $\mathcal{ALCI}$ in [7] which is, however, entirely unsurprising because it concerns a syntactically and semantically restricted case ("rooted CQ answering") where a co-NExpTime bound comes naturally. And second, we believe that the presented upper bound can be extended to the general case where an arbitrary number of roles is allowed, though at the expense of making it considerably more technical.

As usual, we consider conjunctive query entailment instead of CQ answering, i.e., we replace the search problem by its decision problem counterpart. We use the following strategy to obtain a co-NExpTime upper bound for CQ entailment. First, we use a standard technique to show that CQ entailment over unrestricted ABoxes can be reduced to entailment of UCQs (unions of conjunctive queries) over ABoxes that contain only a single individual and no role assertions. More precisely, we use a Turing reduction that requires an exponential number of UCQ entailment checks, where each UCQ contains exponentially many disjuncts in the worst case. Thus, it suffices to establish a co-NExpTime upper bound for each of the required UCQ entailments. Second, we show that if one of the UCQ entailments does not hold, then there is a tree-shaped counter-model with only polynomially many types on each path. Third, we characterize counter-models in terms of tree-interpretations that are annotated in a certain way with subqueries of the original CQ (so-called Q-markings). Thus, we can decide UCQ-(non)-entailment by deciding the existence of a Q-marked tree-interpretation. Fourth, we show that, additionally to the restriction on the number of types, it suffices to consider Q-marked tree-interpretations in which there are only polynomially many different annotations on each path. Finally, we prove that the existence of a Q-marked tree-interpretation with the mentioned restrictions on

the number of types and annotations can be checked by guessing an initial part of the annotated tree-interpretation that has only polynomial depth and thus exponential size, which gives the desired co-NExpTime bound.

2 Preliminaries

We briefly introduce the description logic $\mathcal{S}$, conjunctive queries, and conjunctive query entailment.

Knowledge Bases. We assume standard notation for the syntax and semantics of $\mathcal{S}$ knowledge bases [6]. In particular, $\mathsf{N_C}$ and $\mathsf{N_I}$ are countably infinite and disjoint sets of *concept names* and *individual names*. For the purpose of this paper, we consider a *single transitive role*, denoted throughout by r. *Concepts* are defined inductively: (a) each $A \in \mathsf{N_C}$ is a concept, and (b) if C, D are concepts, then $C \sqcap D$, $\neg C$, and $\exists r.C$ are concepts.[1] A *TBox* is a set of concept inclusions $C \sqsubseteq D$. An *ABox* is a set of *assertions* $C(a)$ and $r(a, b)$. A *knowledge base (KB)* is a pair $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ consisting of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$. We use $\mathcal{I}$ to denote an interpretation, $\Delta^{\mathcal{I}}$ for its domain, and $C^{\mathcal{I}}$ and $r^{\mathcal{I}}$ for the interpretation of a concept C and the role r, respectively. We denote by $\mathsf{Ind}(\mathcal{A})$ the set of all individual names in an ABox $\mathcal{A}$.

Conjunctive Query Entailment. Let $\mathsf{N_V}$ be a countably infinite set of *variables*. A *conjunctive query (CQ or query)* over a KB $\mathcal{K}$ is a finite set of atoms of the form $A(x)$ or $r(x, y)$, where $x, y \in \mathsf{N_V}$, and A is a concept name.[2] For a CQ q over $\mathcal{K}$, let $\mathsf{Var}(q)$ denote the variables occurring in q. A *match for q in an interpretation $\mathcal{I}$* is a mapping $\pi : \mathsf{Var}(q) \to \Delta^{\mathcal{I}}$ such that (i) $\pi(x) \in A^{\mathcal{I}}$ for each $A(x) \in q$, and (ii) $(\pi(x), \pi(y)) \in r^{\mathcal{I}}$ for each $r(x, y) \in q$. We write $\mathcal{I} \models q$ if there is a match for q in $\mathcal{I}$. If $\mathcal{I} \models q$ for every model $\mathcal{I}$ of $\mathcal{K}$, then $\mathcal{K}$ *entails* q, written $\mathcal{K} \models q$. The *query entailment problem* is to decide, given $\mathcal{K}$ and q, whether $\mathcal{K} \models q$. We sometimes also consider *unions of conjunctive queries (UCQs)*, which take the form $\bigcup_i q_i$, where each q_i is a conjunctive query. The notions $\mathcal{I} \models q$ and $\mathcal{K} \models q$ are lifted from CQs to UCQs in the obvious way.

The directed graph G_q associated with a query q is defined as (V, E), where $V = \mathsf{Var}(q)$ and $E = \{(x, y) \mid r(x, y) \in q\}$. When deciding CQ entailment, we assume without loss of generality that the input query q (i.e., the graph G_q) is connected. For $V \subseteq \mathsf{Var}(q)$, we use $q|_{V^{\downarrow}}$ to denote the restriction of q to the set of variables that are reachable in G_q starting from some element in V. We call $q|_{V^{\downarrow}}$ a *proper subquery* of q if it is connected, and use $\mathsf{sub}(q)$ to denote the set of all proper subqueries of q. Obviously, $q \in \mathsf{sub}(q)$.

[1] Concepts of the form $C \sqcup D$ and $\forall r.C$ are viewed as abbreviations.

[2] As usual, individuals in q can be simulated, and queries with answer variables can be reduced to the Boolean CQs considered here.

3 Reduction to Unary ABoxes

The objective of this section is to reduce CQ entailment over arbitrary knowledge bases to UCQ entailment over knowledge bases whose ABoxes contain only a single concept assertion and no role assertions.

Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and q a CQ for which we want to decide whether $\mathcal{K} \models q$. We assume without loss of generality that $\mathcal{T} = \{\top \sqsubseteq C_\mathcal{T}\}$. The announced reduction, which is similar to one used in [5], makes use of the fact that if there is an interpretation $\mathcal{I}$ of $\mathcal{K}$ with $\mathcal{I} \not\models q$, then there is a forest-shaped such model, i.e., a model that consists of an ABox part of unrestricted relational structure and a tree-shaped part rooted at each ABox individual. To check for the existence of a countermodel of this form, we consider all ways in which the query variables can be distributed among the different parts of the model. The query has no match if for each possible distribution, we can select an ABox individual a such that some subquery assigned to the tree model below a is *not* matched in that tree model. This leaves us with the problem of determining the existence of certain tree models (one for each ABox individual) that spoil a (worst-case exponential) set of subqueries.

To formally implement this idea, we require a few preliminary definitions. We use $\mathsf{cl}(\mathcal{K})$ to denote the smallest set that contains $C_\mathcal{T}$, each concept C with $C(a) \in \mathcal{A}$, and is closed under single negation and subconcepts. A *type* is a subset $t \subseteq \mathsf{cl}(\mathcal{K})$ that satisfies the following conditions:

1. $\neg C \in t$ iff $t \not\in C$, for all $\neg C \in \mathsf{cl}(\mathcal{T})$;
2. $C \sqcap D \in t$ iff $C \in t$ and $D \in t$, for all $C \sqcap D \in \mathsf{cl}(\mathcal{T})$;
3. $C_\mathcal{T} \in t$.

We use $\mathsf{tp}(\mathcal{K})$ to denote the set of all types for $\mathcal{K}$. A *completion* of $\mathcal{A}$ is an ABox $\mathcal{A}'$ such that

- $\mathcal{A} \subseteq \mathcal{A}'$ with $\mathsf{Ind}(\mathcal{A}) = \mathsf{Ind}(\mathcal{A}')$;
- for each $a \in \mathsf{Ind}(\mathcal{A})$, we have $\{C \mid C(a) \in \mathcal{A}'\} \in \mathsf{tp}(\mathcal{K})$;
- $r(a,b), r(b,c) \in \mathcal{A}'$ implies $r(a,c) \in \mathcal{A}'$;
- $\exists r.C \in \mathsf{cl}(\mathcal{K})$, $r(a,b) \in \mathcal{A}$, and $C(b) \in \mathcal{A}'$ implies $(\exists r.C)(a) \in \mathcal{A}'$.

We use $\mathsf{cpl}(\mathcal{A})$ to denote the set of all completions for $\mathcal{A}$. A *match candidate* for a completion $\mathcal{A}' \in \mathsf{cpl}(\mathcal{A})$ describes a way of distributing the query variables among the different parts of the model. Formally, it is a mapping $\zeta : \mathsf{Var}(q) \to \{a, a^\downarrow \mid a \in \mathsf{Ind}(\mathcal{A})\}$ such that

- if $A(x) \in q$ and $\zeta(x) = a$, then $A(a) \in \mathcal{A}'$;
- if $r(x,y) \in q$, $\zeta(x) = a$, and $\zeta(y) = b$, then $r(a,b) \in \mathcal{A}'$;
- if $r(x,y) \in q$, $\zeta(x) = a$, $\zeta(y) = b^\downarrow$, and $a \neq b$, then $r(a,b) \in \mathcal{A}'$;
- $r(x,y) \in q$ and $\zeta(x) = a^\downarrow$ implies $\zeta(y) = a^\downarrow$.

For every $r(x,y) \in q$ with $\zeta(x) = a$ and $\zeta(y) = b^\downarrow$ (where potentially $a = b$), define a subset $V \subseteq \mathsf{Var}(q)$ as the smallest set such that

- $y \in V$;

– if $r(x', y') \in q$ with $x' \in V$, then $y' \in V$;
– if $r(x', y') \in q$ with $y' \in V$ and $\zeta(x') = b^{\downarrow}$, then $x' \in V$.

We use $q|_{r(x,y)}$ to denote the restriction of q to the variables in V. Let Q_ζ denote the set of all queries $q|_{r(x,y)}$ obtained in this way. It is straightforward to verify that all these queries are proper subqueries, i.e., $Q_\zeta \subseteq \mathsf{sub}(q)$.

A *query annotation* for $\mathcal{A}'$ identifies the subqueries that do not have a match in the counter-model that we construct. Formally, it is a map $\alpha : \mathsf{Ind}(\mathcal{A}) \to 2^{\mathsf{sub}(q)}$ that satisfies the following conditions:

1. for every match candidate ζ for $\mathcal{A}'$, there is a query $q|_{r(x,y)} \in Q_\zeta$ such that $q|_{r(x,y)} \in \alpha(a)$ where $\zeta(y) = a^{\downarrow}$;
2. $q \in \alpha(a)$ for all $a \in \mathsf{Ind}(\mathcal{A})$.

For each $a \in \mathsf{Ind}(\mathcal{A})$, we use $\mathcal{A}'|_a$ to denote the restriction of $\mathcal{A}'$ to assertions of the form $C(a)$. The proof of the following lemma is similar to that of a closely related result in [6].

Lemma 1. *$\mathcal{K} \not\models q$ iff there is a completion $\mathcal{A}'$ of $\mathcal{A}$ and a query annotation α for $\mathcal{A}'$ such that for all $a \in \mathsf{Ind}(\mathcal{A})$, we have $\mathcal{K}_a \not\models \bigcup \alpha(a)$, where $\mathcal{K}_a = (\mathcal{T}, \mathcal{A}'|_a)$.*

Lemma 1 constitutes the announced reduction: to decide whether $\mathcal{K} \models q$, we can enumerate all completions $\mathcal{A}'$ of $\mathcal{A}$ and query annotations α for $\mathcal{A}'$, and then perform the required UCQ entailment checks.

4 Characterization of Counter-models

It remains to decide whether $\mathcal{K}_a \models \bigcup \alpha(a)$ holds for each $a \in \mathsf{Ind}(\mathcal{A})$. Since $\alpha(a)$ may contain exponentially many different subqueries of q (this is what actually happens in the lower bound proved in [5]), it is challenging to do this in co-NExpTime. We start with a characterization of counter-models. In the remainder of the section, for readability, we fix some $a \in \mathsf{Ind}(\mathcal{A})$, and we use Q to denote $\alpha(a)$ and C_a to denote $\bigsqcap\{C \mid C(a) \in \mathcal{A}'\}$.

Many of the subsequent techniques and results will be concerned with trees and tree interpretations, which we introduce next. Let Σ be an arbitrary set. Then a *tree (over Σ with root p)* is a set $T = \{p \cdot w \mid w \in S\}$ where $p \in \Sigma^*$ and $S \subseteq \Sigma^*$ is a prefix-closed set of words. Each node $w \cdot c \in T$, where $w \in T$ and $c \in \Sigma$, is a *child* of w. For a node $w \in T$, $|w|$ denotes the length of w, disregarding the prefix p (so that the root of T has length 0). We say *the branching degree of T is bounded by k* if $|\{c \in \Sigma \mid w \cdot c \in T\}| \leq k$ for all $w \in T$. A *path in T*, is a (potentially infinite) sequence $w_0, w_1, \ldots$ of elements from T such that (i) w_0 is the root of T, and (ii) for each $i > 0$, w_i is a child of w_{i-1}. If T is a tree and $f : T \to S$ is a function with S finite, then we use $\mathsf{max}(T, f)$ to denote the maximal number of distinct values that f can take on an arbitrary path in T.

An interpretation $\mathcal{I}$ is a *tree interpretation* if $\Delta^{\mathcal{I}}$ is a tree. We introduce the notation $\mathsf{root}(\mathcal{I})$ to denote the root of the tree $\Delta^{\mathcal{I}}$. A tree interpretation $\mathcal{I}$ is a *tree model* of $\mathcal{K}_a$ if

- $\mathcal{I}$ is a model of $\mathcal{T}$, and $\mathsf{root}(\mathcal{I}) \in C_a^{\mathcal{I}}$,
- $r^{\mathcal{I}} = \{(w, w \cdot c) \mid w, w \cdot c \in \Delta^{\mathcal{I}} \wedge c \in \Sigma\}^+$, and
- for all $\exists r.C \in \mathsf{cl}(\mathcal{K})$ and $w \in (\exists r.C)^{\mathcal{I}}$, there is $c \in \Sigma$ such that $w \cdot c \in C^{\mathcal{I}}$, i.e., all relevant existential restrictions are satisfied in one step.

Given a tree interpretation $\mathcal{I}$ and $w \in \Delta^{\mathcal{I}}$, we use $\mathcal{I}|_w$ to denote the restriction of $\mathcal{I}$ to the subtree rooted at w.

The following lemma shows that we can restrict our attention to tree-shaped interpretations in which only polynomially many types appear on any given path. As the proof of the lemma is surprisingly subtle, we defer it to the appendix of a longer version of this submission [1]. Given an interpretation $\mathcal{I}$, we use $t_{\mathcal{I}}(w)$ to refer to the type of $w \in \Delta^{\mathcal{I}}$ in $\mathcal{I}$, i.e. $\{C \in \mathsf{cl}(\mathcal{K}) \mid w \in C^{\mathcal{I}}\}$.

Lemma 2. *If $\mathcal{K}_a \not\models \bigcup Q$, then there is an interpretation $\mathcal{I}$ such that:*

1. *$\mathcal{I}$ is a tree model of $\mathcal{K}_a$, and $\mathcal{I} \not\models \bigcup Q$, and*
2. *$\mathsf{max}(\Delta^{\mathcal{I}}, t_{\mathcal{I}}) \leq |\mathsf{cl}(\mathcal{K})|$.*

To characterize counter-models, we employ *marking* of interpretations, similar to that in [5]. A marking simulates a top-down walk through a tree interpretation $\mathcal{I}$ greedily matching the variables of the queries in Q. The marking fails if we arrive at a subquery that is fully matched along this walk. As we show next, the existence of a marking for a tree interpretation $\mathcal{I}$ is a necessary and sufficient condition for $\mathcal{I} \not\models \bigcup Q$.

For a query p and a variable $x \in \mathsf{Var}(p)$, we say that x is *consumed (in p)* by a type t if $\{A \mid A(x) \in p\} \subseteq t$ and $\{y \mid r(y, x) \in p\} = \emptyset$. Given a type $t \in \mathsf{tp}(\mathcal{K})$ and a query $p \in \mathsf{sub}(q)$, we denote by $\mathsf{sub}^t(p)$ the set of all proper subqueries of p^t, where p^t is obtained from p by removing all atoms involving a variable that is consumed by t. In other words, $\mathsf{sub}^t(p)$ is the set of connected components in the reduced query p^t. Trivially, $\mathsf{sub}^t(p) = \{p\}$ if t does not consume any variable in p.

The following lemma describes a single step of the top-down walk through a tree interpretation.

Lemma 3. *Assume a tree interpretation $\mathcal{I}$, $w \in \Delta^{\mathcal{I}}$ and any set P of queries. Then $\mathcal{I}|_w \not\models \bigcup P$ iff there is a set P' such that:*

(i) P' contains some non-empty $p' \in \mathsf{sub}^{t_{\mathcal{I}}(w)}(p)$ for each $p \in P$;
(ii) $\mathcal{I}|_{w'} \not\models \bigcup P'$ for each child w' of w in $\Delta^{\mathcal{I}}$.

Proof. For the if direction, we show that if $\mathcal{I}|_w \models \bigcup P$, then there is no set P' satisfying (i) and (ii). If $\mathcal{I}|_w \models \bigcup P$, then there is a match π in $\mathcal{I}|_w$ for some $p \in P$. We show that then, for each $p' \in \mathsf{sub}^{t_{\mathcal{I}}(w)}(p)$, there exists a child w' of w such that $\mathcal{I}|_{w'}$ admits a match for p'. This implies that there is no set P', since there is no possible choice of a subquery in $\mathsf{sub}^{t_{\mathcal{I}}(w)}(p)$ to be included.

Let π be a match for p in $\mathcal{I}|_w$, and let $\mathsf{sub}^{\pi(w)}(p)$ denote the set of all proper subqueries of the query $p^{\pi(w)}$ that results from p by dropping each atom involving a variable x with $\pi(x) = w$. By definition of a match, each $x \in \mathsf{Var}(p)$ with

$\pi(x) = w$ is consumed by $t_{\mathcal{I}}(w)$. This implies that all atoms removed from p to obtain $p^{\pi(w)}$ are also removed to obtain $p^{t_{\mathcal{I}}(w)}$, and thus each $p' \in \mathsf{sub}^{t_{\mathcal{I}}(w)}(p)$ is contained in some $p'' \in \mathsf{sub}^{\pi(w)}(p)$. Since π is a match for p, each $p'' \in \mathsf{sub}^{\pi(w)}(p)$ has a match in $\mathcal{I}|_{w'}$ for some child w' of w (in particular, π restricted to the domain of $\mathcal{I}|_{w'}$ is such a match), and so does each $p' \subseteq p''$. This shows that, for each $p' \in \mathsf{sub}^{t_{\mathcal{I}}(w)}(p)$, there exists a child w' of w such that $\mathcal{I}|_{w'} \models p'$.

For the other direction we show that if there does not exist a set P' as above, then $\mathcal{I}|_w \models \bigcup P$. Assume that there is no P' satisfying (i) and (ii). Then we can select some $p \in P$ such that for each non-empty $p' \in \mathsf{sub}^{t_{\mathcal{I}}(w)}(p)$, there is a child w' of w with $\mathcal{I}|_{w'} \models p'$, and we can select a match $\pi_{p'}$ in $\mathcal{I}|_{w'}$ for each p'. Observe that each $x \in \mathsf{Var}(p)$ that is not consumed by $t_{\mathcal{I}}(w)$ occurs in some p' and is in the scope of some $\pi_{p'}$. It can be easily verified that a match π for p can be composed by taking the union of all π'_p, and setting $\pi(x) = w$ for all remaining variables x. This shows $\mathcal{I}|_w \models p$ and $\mathcal{I}|_w \models \bigcup P$. ❑

We can now formally define the notion of a marking, which describes a top-down walk through a whole tree interepretation.

Definition 1. *Let $\mathcal{I}$ be a tree interpretation. A* Q-marking *for $\mathcal{I}$ is a mapping $\mu : \Delta^{\mathcal{I}} \to 2^{\mathsf{sub}(q)}$ such that:*

1. $\mu(\mathsf{root}(\mathcal{I})) = Q$,
2. *for each $w \in \Delta^{\mathcal{I}}$ and each pair $w \cdot i, w \cdot j \in \Delta^{\mathcal{I}}$, $\mu(w \cdot i) = \mu(w \cdot j)$,*
3. *for each $w \cdot i \in \Delta^{\mathcal{I}}$, $\mu(w \cdot i)$ is a set containing a non-empty $p' \in \mathsf{sub}^{t_{\mathcal{I}}(w)}(p)$ for each $p \in \mu(w)$.*

Using Lemma 3, we can characterize query non-entailment as follows:

Lemma 4. *There is a Q-marking for a tree interpretation $\mathcal{I}$ iff $\mathcal{I} \not\models \bigcup Q$.*

Proof. For the if direction, assume $\mathcal{I} \not\models \bigcup Q$. We define a Q-marking μ for $\mathcal{I}$ inductively:

- $\mu(\mathsf{root}(\mathcal{I})) = Q$,
- $\mu(w{\cdot}c) = \mu(w)'$ for all $w{\cdot}c \in \Delta^{\mathcal{I}}$, where $\mu(w)'$ is a $\subseteq$-minimal set of subqueries satisfying conditions (i) and (ii) of Lemma 3 (where we take $P = \mu(w)$ and $P' = \mu(w)'$).

Note that a suitable set $\mu(\mathsf{root}(\mathcal{I}))'$ exists for the children of the root because $\mathcal{I} \not\models \bigcup Q$. Then at each step $w \cdot c$, condition (ii) in Lemma 3 ensures that $\mathcal{I}|_{w \cdot c} \not\models \bigcup \mu(w \cdot c)$. Applying the lemma again we ensure the existence of a suitable set $\mu(w \cdot c)'$ for the children of $w \cdot c$. It is trivial to verify that μ satisfies the conditions in the definition of Q-marking (in particular, for condition 3 we use condition (i) in Lemma 3).

The other direction follows easily from the first condition in Definition 1, which ensures that the root is always marked with Q, and the following claim:

($*$) If μ is a Q-marking for $\mathcal{I}$, then $\mathcal{I}|_w \not\models \bigcup \mu(w)$ for every $w \in \Delta^{\mathcal{I}}$.

To show (∗), we assume for a contradiction that μ is a Q-marking and that $\mathcal{I}|_w \models \bigcup \mu(w)$ for some $w \in \Delta^{\mathcal{I}}$. That is, $\mathcal{I}|_w \models p$ for some $p \in \mu(w)$. Among all such pairs (w, p), we select one with minimal $|\mathsf{Var}(p)|$, i.e., such that $|\mathsf{Var}(p)| \leq |\mathsf{Var}(p')|$ for every $w' \in \Delta^{\mathcal{I}}$ and every $p' \in \mu(w')$ such that $\mathcal{I}|_{w'} \models p'$. In the case where $t_{\mathcal{I}}(w)$ consumes no variable in p, we have that for every child w' of w, $\mu(w) = \mu(w')$ and $\mathcal{I}|_w \models p$ iff $\mathcal{I}|_{w'} \models p$. We can iteratively apply this argument to choose a $w^* \in \Delta^{\mathcal{I}|_w}$ (either w itself or a first descendant where some variable is consumed) such that $t_{\mathcal{I}}(w^*)$ consumes some $x \in \mathsf{Var}(p)$, $\mathcal{I}|_{w^*} \models p$, and $\mu(w^*) = \mu(w)$. The fact that $t_{\mathcal{I}}(w^*)$ consumes some $x \in \mathsf{Var}(p)$ ensures $|\mathsf{Var}(p')| < |\mathsf{Var}(p)|$ for every $p' \in \mathsf{sub}^{t_{\mathcal{I}}(w^*)}(p)$. Since μ is a Q-marking for $\mathcal{I}$ and $p \in \mu(w^*)$, by conditions 2 and 3 in Definition 1, there must be some non-empty $p' \in \mathsf{sub}^{t_{\mathcal{I}}(w^*)}(p)$ such that $p' \in \mu(w')$ for all children w' of w^*. We know from Lemma 3 that $\mathcal{I}|_{w^*} \models \{p\}$ implies that $\mathcal{I}|_{w'} \models \{p'\}$ for some child w' of w^*. But as $|\mathsf{Var}(p')| < |\mathsf{Var}(p)|$, this is a contradiction. ❑

We have shown that UCQ non-entailment reduces to deciding the existence of a marking. The following lemma will help us to show that the latter problem can be decided in NExpTime. It shows that, even though there can be exponentially many queries in Q, the query set changes only a few times on each path of a marked interpretation. More precisely:

Lemma 5. *If $\mathcal{I} \not\models \bigcup Q$, then $\mathcal{I}$ admits a Q-marking μ with $\mathsf{max}(\Delta^{\mathcal{I}}, \mu) \leq |\mathsf{Var}(q)|^2 + 1$.*

Proof. Let μ be the Q-marking defined in the proof of Lemma 4. We consider an arbitrary path $w_1, w_2, \ldots$ in $\mathcal{I}$, and show that $l = |\{\mu(w_1), \mu(w_2), \ldots\}| \leq |\mathsf{Var}(q)|^2 + 1$. We let $J = \{i \mid \mu(w_i) \neq \mu(w_{i+1})\}$. We will show that $|J| \leq |q|^2$. The desired bound will follow from this and the fact that $l \leq |J| + 1$. Let $t_i = t_{\mathcal{I}}(w_i)$ for all $i \geq 0$. We say a query q' is *i-matched* if q' has a match in $\mathcal{I}_i$ but not on $\mathcal{I}_{i-1}$, where $\mathcal{I}_k$ is defined by setting (i) $\Delta^{\mathcal{I}_k} = \{(1, t_1), \ldots, (k, t_k)\}$; (ii) $r^{\mathcal{I}_k} = \{((i, t_i), (j, t_j)) \mid j > i\}$; (iii) $A^{\mathcal{I}_k} = \{(i, t_i) \mid A \in t_i\}$ for all $A \in \mathsf{N_C}$. Note that, for any query q', there is at most one index i such that q' is i-matched. For each pair $x, y \in \mathsf{Var}(q)$, let $q|^{x,y}$ be the query that is obtained by restricting $q|_{\{x\}\downarrow}$ to the variable y and the variables that reach y in the graph G_q. Let $X = \{q|^{x,y} \mid x, y \in \mathsf{Var}(q)\}$. Note that $|X| \leq |\mathsf{Var}(q)|^2$. We now show that for each $i \in J$, there exists some $q' \in X$ such that q' is i-matched. Since there is at most one i for each q', this implies $|J| \leq |X| \leq |q|^2$ and the bound follows.

Consider an arbitrary $i \in J$. Then $\mu(w_i) \neq \mu(w_{i+1})$ implies that for some $p' \in \mu(w_i)$, $\mu(w_{i+1})$ contains some $p'' \neq p'$ from $\mathsf{sub}^{t_{\mathcal{I}}(w)}(p')$, and some $x \in \mathsf{Var}(p')$ is consumed by $t_{\mathcal{I}}(w_i)$. By definition, the query p' is a proper subquery of some $p \in Q$. Observe that, if we restrict our attention to p and its subqueries, the marking μ 'moves' to a strictly smaller subquery at every type that consumes some variable. Let M be the set of source variables in the query graph G_p of this p, i.e. $M = \{y \in \mathsf{Var}(p) \mid \{y' \mid r(y', y) \in p\} = \emptyset\}$. It is not hard to see that, if $x \in \mathsf{Var}(p')$ is consumed by $t_{\mathcal{I}}(w_i)$, each $q|^{y,x}$ with $y \in M$ has a match in $\mathcal{I}_i$. To see that there exists at least one $y \in M$ such that $q|^{y,x}$ is i-matched, assume towards a contradiction that there is some $j < i$ such that each $q|^{y,x}$ has a match

in $\mathcal{I}_j$, and take the smallest such j. Then all variables that reach x in G_q are consumed by some type on the path to w_j, and w_j is marked with some $p'' \subseteq p$ where $\{y \mid r(y,x) \in p''\} = \emptyset$. As x is consumed by $t_{\mathcal{I}}(w_j)$, then the markings of all descendants of w_j contain some subquery of p'' where x does not occur. This contradicts the fact that $p' \in \mu(w_i)$ and $x \in \mathsf{Var}(p')$. ❑

As a direct consequence of Lemmas 2, 4 and 5, we obtain the following characterization of counter-models; this is the basis of our UCQ entailment algorithm.

Theorem 1. *$\mathcal{K}_a \not\models \bigcup Q$ iff there is a tree interpretation $\mathcal{I}$ such that:*

(A) $\mathcal{I}$ is a model of $\mathcal{K}_a$ with $\mathsf{max}(\Delta^{\mathcal{I}}, t_{\mathcal{I}}) \leq |\mathsf{cl}(\mathcal{K})|$;
(B) $\mathcal{I}$ admits some Q-marking μ and $\mathsf{max}(\Delta^{\mathcal{I}}, \mu) \leq |\mathsf{Var}(q)|^2 + 1$.

By removing domain elements not needed to satisfy existential restrictions from $\mathsf{cl}(\mathcal{K})$, it is standard to show that we can assume the interpretation $\mathcal{I}$ from Theorem 1 to have branching degree at most $|\mathsf{cl}(\mathcal{K})|$.

5 Witnesses of Counter-models

By Theorem 1, $\mathcal{K}_a \not\models \bigcup Q$ can be decided by checking whether there is a tree interpretation that satisfies conditions (A) and (B). As we show next, the existence of such an interpretation $\mathcal{I}$ is guaranteed if we can find an initial part of $\mathcal{I}$ whose depth is bounded by $d_{\mathcal{K},q} := |\mathsf{cl}(\mathcal{K})| \times (|\mathsf{Var}(q)|^2 + 1)$. Since the branching degree of $\mathcal{I}$ is linear in the size of $\mathcal{K}$, this initial part is of at most exponential size. A nondeterministic exponential time procedure for checking $\mathcal{K}_a \not\models \bigcup Q$ is then almost immediate. We represent initial parts of countermodels as follows.

Definition 2. *A* witness for "$\mathcal{K}_a \not\models \bigcup Q$" *is a node-labeled tree $W = (T, \tau, \rho)$ where $\tau : T \to \mathsf{tp}(\mathcal{K})$ and $\rho : T \to 2^{\mathsf{sub}(q)}$, such that:*

1. *The branching degree of T is bounded by $|\mathsf{cl}(\mathcal{K})|$.*
2. *For each $w \in T$, $|w| \leq d_{\mathcal{K},q}$.*
3. *$\mathsf{max}(T, \tau) \leq |\mathsf{cl}(\mathcal{K})|$ and $\mathsf{max}(T, \rho) \leq |\mathsf{Var}(q)|^2 + 1$;*
4. *$\{C \mid C(a) \in \mathcal{A}'\} \subseteq \tau(e)$ and $\rho(e) = Q$ for the root e of T.*
5. *For all $w \in T$ with $|w| < d_{\mathcal{K},q}$ and $\exists r.C \in \tau(w)$, there is a child w' of w with $C \in \tau(w')$.*
6. *For each $w \in T$ and each child w' of w, $\neg\exists r.D \in \tau(w)$ implies $\{\neg D, \neg\exists r.D\} \subseteq \tau(w')$.*
7. *For each pair w_1, w_2 of children of w, $\rho(w_1) = \rho(w_2)$ is a set containing some nonempty $p' \in \mathsf{sub}^t(p)$ for each $p \in \rho(w)$.*

An initial part of a tree interpretation represented by a witness can be unravelled into a tree interpretation that satisfies (A) and (B) of Theorem 1, thus witnessing $\mathcal{K}_a \not\models \bigcup Q$.

Theorem 2. *$\mathcal{K}_a \not\models \bigcup Q$ iff there exists a witness W for "$\mathcal{K}_a \not\models \bigcup Q$".*

Proof. For the 'only if' direction, by Theorem 1 there exists a tree-model $\mathcal{I}$ of $\mathcal{K}_a$ and a Q-marking μ for $\mathcal{I}$ such that $\mathsf{max}(\Delta^{\mathcal{I}}, t_{\mathcal{I}}) \leq |\mathsf{cl}(\mathcal{K})|$, $\mathsf{max}(\Delta^{\mathcal{I}}, \mu) \leq |\mathsf{Var}(q)|^2 + 1$, and the branching degree of $\mathcal{I}$ is at most $|\mathsf{cl}(\mathcal{K})|$. We can obtain a witness by restricting $\mathcal{I}$ and μ to the first $d_{\mathcal{K},q}$ levels. More precisely, $W = (T, \tau, \rho)$ is obtained by setting:

- $T = \{w \in \Delta^{\mathcal{I}} \mid |w| \leq d_{\mathcal{K},q}\}$;
- $\tau(w) = t_{\mathcal{I}}(w)$ and $\rho(w) = \mu(w)$ for all $w \in T$.

For the other direction, observe that a witness $W = (T, \tau, \rho)$ is almost a Q-marked model of $\mathcal{K}_a$, except a node $w \in T$ with $|w| = d_{\mathcal{K},q}$ may not have the children it needs to satisfy the existential restrictions. However, since the path from the root to w has $d_{\mathcal{K},q} + 1$ nodes and due to (3) in Definition 2, there exists a pair of nodes on this path that share the same type and query set. This allows us to obtain a tree-model and a Q-marking by unraveling W as follows.

For each node $w \in T$, let $s(w)$ be the shortest prefix of w such that $\tau(s(w)) = \tau(w)$ and $\rho(s(w)) = \rho(w)$. Let $D \subseteq T^*$ be the smallest set of such that:

- the root of T belongs to D, and
- if $w_0 \cdots w_n \in D$, then $w_0 \cdots w_n w \in D$ for all children w of $s(w_n)$.

Consider the following interpretation $\mathcal{I}$ and marking μ:

- $\Delta^{\mathcal{I}} = D$;
- $A^{\mathcal{I}} = \{w_0 \cdots w_n \in \Delta^{\mathcal{I}} \mid A \in \tau(v_n)\}$ for all concept names A;
- $r^{\mathcal{I}} = \{(w_0 \cdots w_{n-1}, w_0 \cdots w_n) \mid w_0 \cdots w_n \in \Delta^{\mathcal{I}}\}$;
- $\mu(w_0 \cdots w_n) = \rho(w_n)$ for all $w_0 \cdots w_n \in \Delta^{\mathcal{I}}$.

It is easy to check that μ is a Q-marking for $\mathcal{I}$. To see that $\mathcal{I}$ is model of $\mathcal{K}_a$, observe that for each node $w \in T$ with $|w| = d_{\mathcal{K},q}$, there is a proper prefix w' of w such that $s(w') \neq w'$. This means that such a w will never be added to a path in $\Delta^{\mathcal{I}}$. This implies that each $w_0 \cdots w_n \in \Delta^{\mathcal{I}}$ has $|w_n| < d_{\mathcal{K},q}$ and hence satisfies all the existential restrictions. ❑

We can check for the existence of a witness by nondeterministically guessing an (exponential size) candidate structure $W = (T, \tau, \rho)$ and then verifying conditions (1-7) in Definition 2. The latter is feasible in time exponential in $|\mathcal{K}|$ and $|q|$. Hence, $\mathcal{K}_a \not\models \bigcup Q$ can be decided nondeterministically in time exponential in $|\mathcal{K}|$ and $|q|$.

For the overall algorithm, observe that each completion $\mathcal{A}'$ of $\mathcal{A}$ is of size polynomial in $|\mathcal{K}|$ and $|q|$, while the size of $\alpha(a)$ is at most exponential in $|\mathcal{K}|$ and $|q|$ for each $a \in \mathsf{Ind}(\mathcal{A})$. Thus, using Lemma 1, checking $\mathcal{K} \not\models q$ is trivially in NExpTime provided that checking $\mathcal{K}_a \not\models \bigcup \alpha(a)$ is NExpTime. By combining this with the matching lower bound in [5], we get:

Theorem 3. *CQ entailment over $\mathcal{S}$ KBs with one transitive role, and no other roles, is* co-NExpTime*-complete.*

6 Conclusion

We believe that Theorem 3 can be extended to the case where there is an arbitrary number of roles, both transitive and unrestricted ones. This requires the combination of the techniques presented in this paper with the ones developed in [5]. In particular, different roles used in a query $p \in Q$ induce a partitioning of p into different "clusters", and each cluster can be treated in a similar way as an entire, unpartitioned query $p \in Q$ in the current paper. Since the technical details, which we are currently working out, can be expected to become somewhat cumbersome, we believe that it is instructive to first concentrate on the case of a single transitive role as we have done in this paper.

It is interesting to note that the techniques from this paper can be used to reprove in a transparent way the ExpTime upper bound for CQ answering over $\mathcal{S}$ knowledge bases that contain only a single concept assertion and no role assertions from [5]—restricted to a single transitive role, of course. In the case of such ABoxes, we do not need the machinery from Sections 3 and 5, nor the (subtle to prove) Lemma 2. The essential technique is Q-markings, which can be simplified to maps from $\Delta^{\mathcal{I}}$ to $\mathsf{sub}(q)$ instead of to $2^{\mathsf{sub}(q)}$ because Q is a singleton that consists only of the input query. By Lemma 4, it suffices to check for the existence of a tree-shaped interpretation $\mathcal{I}$ along with a Q-marking for $\mathcal{I}$. This can be done by a standard type-elimination procedure.

References

1. Meghyn Bienvenu, Thomas Eiter, Magdalena Ortiz, and Mantas Šimkus. Query answering in the description logic $\mathcal{S}$. Technical Report INFSYS RR-1843-10-01 (available at `http://www.kr.tuwien.ac.at/research/reports/`), 2010.
2. Craig Boutilier, editor. *IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009*, 2009.
3. Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The DL-Lite family. *J. of Automated Reasoning*, 39(3):385–429, 2007.
4. Diego Calvanese, Giuseppe De Giacomo, and Maurizio Lenzerini. On the decidability of query containment under constraints. In *Proc. of the 17th ACM SIGACT SIGMOD SIGART Symp. on Principles of Database Systems (PODS'98)*, pages 149–158, 1998.
5. Thomas Eiter, Carsten Lutz, Magdalena Ortiz, and Mantas Šimkus. Query answering in description logics with transitive roles. In Boutilier [2], pages 759–764.
6. Birte Glimm, Carsten Lutz, Ian Horrocks, and Ulrike Sattler. Answering conjunctive queries in the $\mathcal{SHIQ}$ description logic. *Journal of Artificial Intelligence Research*, 31:150–197, 2008.
7. Carsten Lutz. The complexity of conjunctive query answering in expressive description logics. In Alessandro Armando, Peter Baumgartner, and Gilles Dowek, editors, *Proceedings of the 4th International Joint Conference on Automated Reasoning (IJCAR2008)*, number 5195 in LNAI, pages 179–193. Springer, 2008.

8. Carsten Lutz, David Toman, and Frank Wolter. Conjunctive query answering in the description logic EL using a relational database system. In Boutilier [2], pages 2070–2075.
9. Magdalena Ortiz, Diego Calvanese, and Thomas Eiter. Data complexity of query answering in expressive description logics via tableaux. *J. of Automated Reasoning*, 41(1):61–98, 2008. doi:10.1007/s10817-008-9102-9. Preliminary version available as Tech.Rep. INFSYS RR-1843-07-07, Institute of Information Systems, TU Vienna, Nov. 2007.
10. Ulrike Sattler. Description logics for the representation of aggregated objects. In W. Horn, editor, *Proceedings of the Fourteenth European Conference on Artificial Intelligence (ECAI'00)*. IOS Press, Amsterdam, 2000.
11. Sergio Tessaris. *Questions and Answers: Reasoning and Querying in Description Logic*. PhD thesis, University of Manchester, Department of Computer Science, April 2001.